An introduction to Psychometric Theory with applications in R

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

February, 2013

Overview

1. Overview Psychometrics and R
   - What is Psychometrics
   - What is R
2. Part I: an introduction to R
   - What is R
   - A brief example
   - Basic steps and graphics
3. Day 1: Theory of Data, Issues in Scaling
4. Day 2: More than you ever wanted to know about correlation
5. Day 3: Dimension reduction through factor analysis, principal components analyze and cluster analysis
6. Day 4: Classical Test Theory and Item Response Theory
7. Day 5: Structural Equation Modeling and applied scale construction

Outline of Day 1/part 1

1. What is psychometrics?
    - Conceptual overview
    - Theory: the organization of Observed and Latent variables
    - A latent variable approach to measurement
    - Data and scaling
    - Structural Equation Models
2. What is R? Where did it come from, why use it?
    - Installing R on your computer and adding packages
    - Installing and using packages
    - Implementations of R
    - Basic R capabilities: Calculation, Statistical tables, Graphics
    - Data sets
3. Basic statistics and graphics
    - 4 steps: read, explore, test, graph
    - Basic descriptive and inferential statistics
4. TOD

What is psychometrics?

In physical science a first essential step in the direction of learning any subject is to find principles of numerical reckoning and methods for practicably measuring some quality connected with it. I often say that when you can measure what you are speaking about and express it in numbers you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the stage of science, whatever the matter may be. (Thomsom, 1891)

Taken from Michell (2003) in his critique of psychometrics: Michell, J. The Quantitative Imperative: Positivism, NaȚve Realism and the Place of Qualitative Methods in Psychology, Theory & Psychology, Vol. 13, No. 1, 5-31 (2003)

What is psychometrics?

The character which shapes our conduct is a definite and durable 'something', and therefore . . . it is reasonable to attempt to measure it. (Galton, 1884)

The history of science is the history of measurement" (J. M. Cattell, 1893)

Whatever exists at all exists in some amount. To know it thoroughly involves knowing its quantity as well as its quality (E.L. Thorndike, 1918)

What is psychometrics?

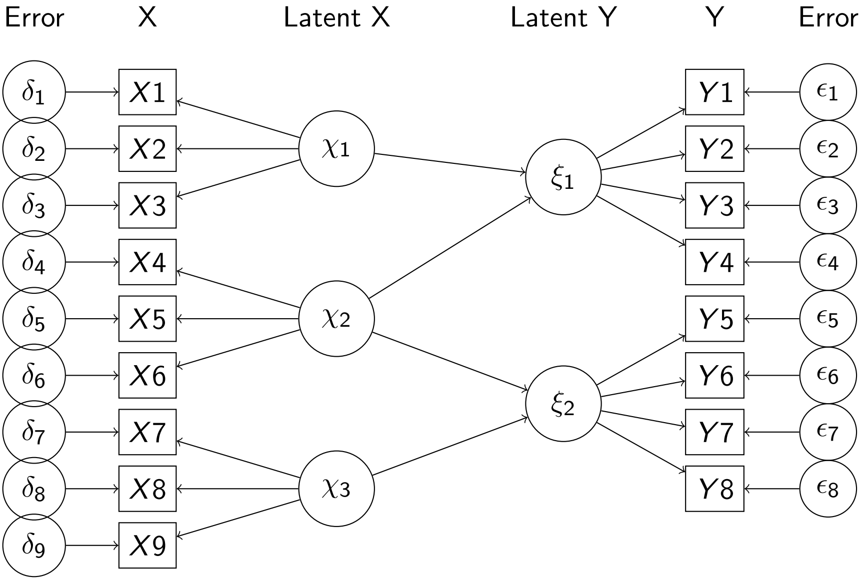
> *We hardly recognize a subject as scientific if measurement is not one of its tools (Boring, 1929)*

> *There is yet another [method] so vital that, if lacking it, any study is thought ... not be scientific in the full sense of the word. This further an crucial method is that of measurement. (Spearman, 1937)*

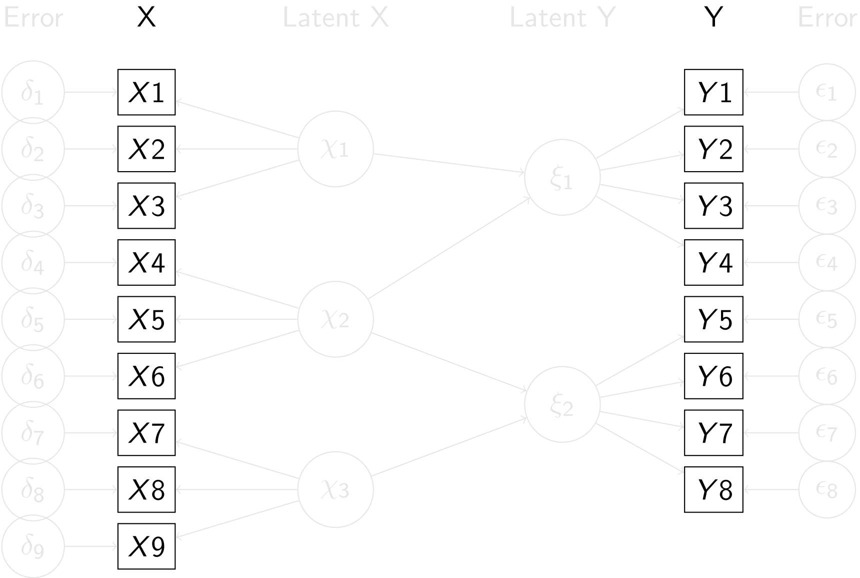
> *One's knowledge of science begins when he can measure what he is speaking about and express in numbers (Eysenck, 1973)*

Psychometrics: the assigning of numbers to observed psychological phenomena and to unobserved concepts. Evaluation of the fit of theoretical models to empirical data.

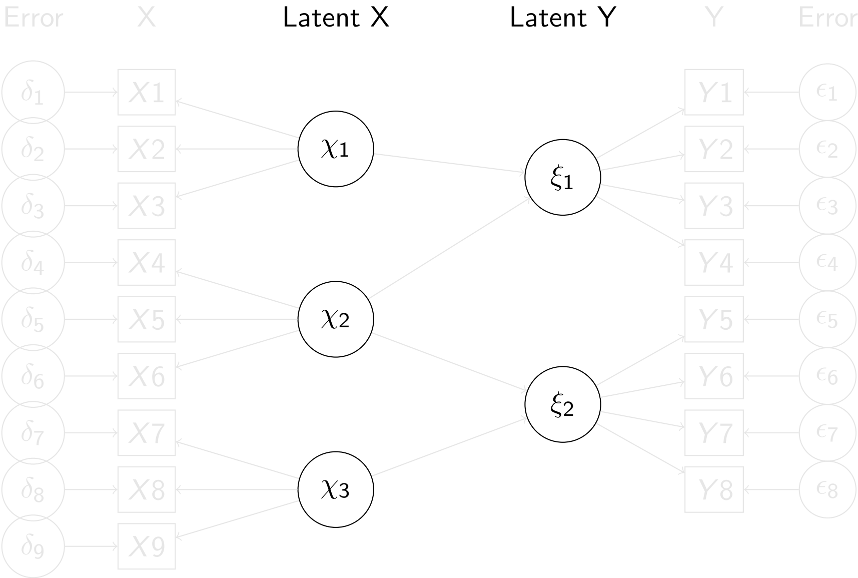
Psychometric Theory: A conceptual Syllabus



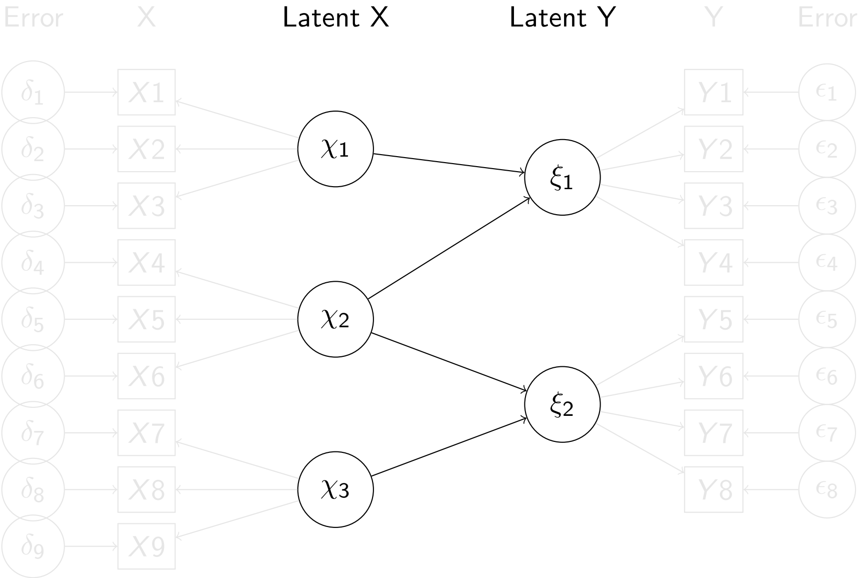
Observed Variables



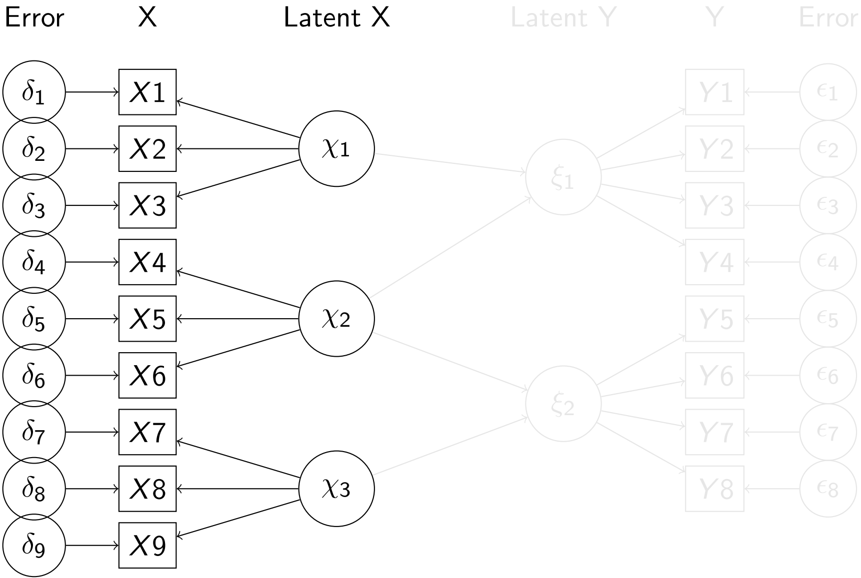
Latent Variables



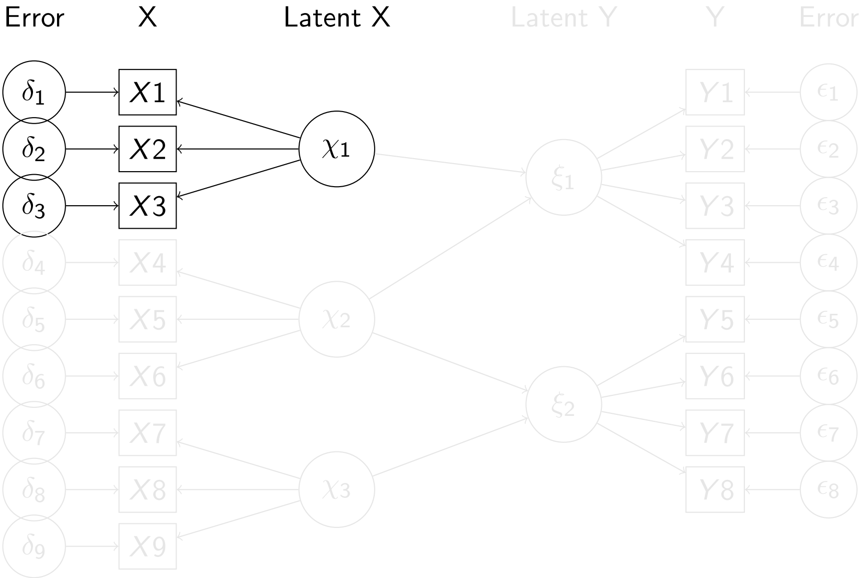
Theory


Error
X
Latent X
Latent Y
Y
Error
δ_1
δ_2
δ_3
δ_4
δ_5
δ_6
δ_7
δ_8
δ_9
X1
X2
X3
X4
X5
X6
X7
X8
X9
χ_1
χ_2
χ_3
ξ_1
ξ_2
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
ϵ_1
ϵ_2
ϵ_3
ϵ_4
ϵ_5
ϵ_6
ϵ_7
ϵ_8

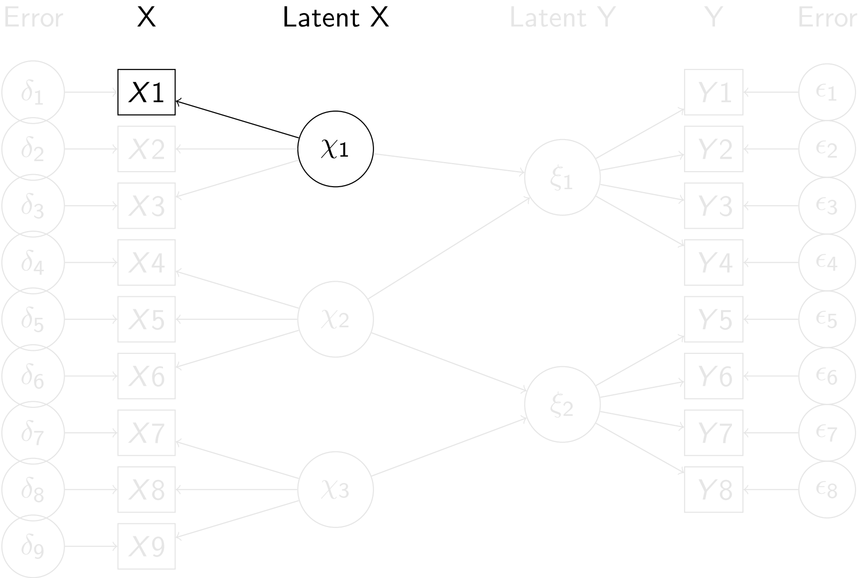
Measurement: A latent variable approach.



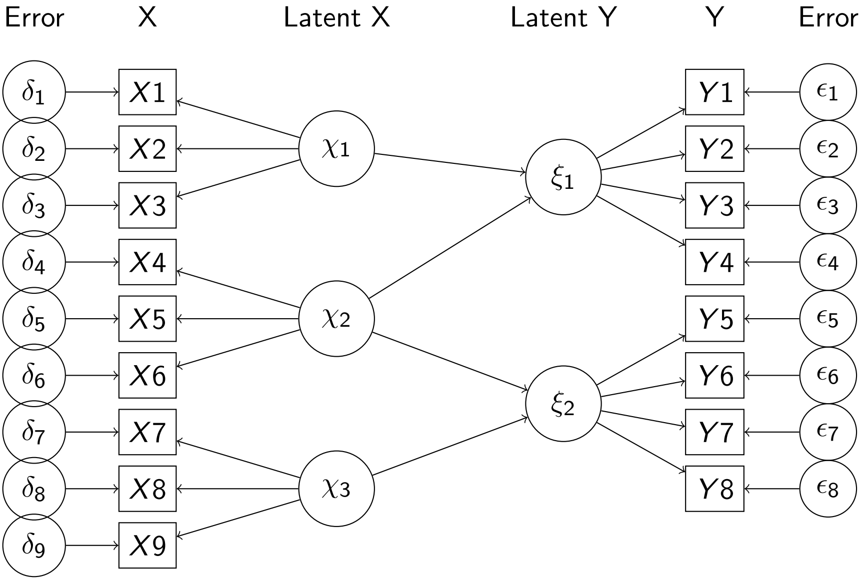
Reliability: How well does a test reflect one latent trait?



A theory of data and fundamentals of scaling



Psychometric Theory: Data, Measurement, Theory



Use R

R: Statistics for all of us

1. What is it?
2. Why use it?
3. Common (mis)perceptions of R
4. Examples for psychologists
   - graphical displays
   - basic statistics
   - advanced statistics
   - Although programming is easy in R, that is beyond the scope of today

R: What is it?

1. R: An international collaboration
2. R: The open source - public domain version of S+
3. R: Written by statistician (and all of us) for statisticians (and the rest of us)
4. R: Not just a statistics system, also an extensible language.
   - This means that as new statistics are developed they tend to appear in R far sooner than elsewhere.
   - R facilitates asking questions that have not already been asked.
5. R: encourages publications of "Reproducible Research"
   - integrate data, code, text into one document
   - Sweave and knitr

Statistical Programs for Psychologists

- General purpose programs
    - R
    - S+
    - SAS
    - SPSS
    - STATA
    - Systat
- Specialized programs
    - Mx
    - EQS
    - AMOS
    - LISREL
    - MPlus
    - Your favorite program

Statistical Programs for Psychologists

- General purpose programs
  - R
  - $+
  - A
  - $P$$
  - $TATA
  - $y$tat
- Specialized programs
  - Mx (OpenMx is part of R)
  - EQ$
  - AMO$
  - LI$REL
  - MPlu$
  - Your favorite program

R: A way of thinking

- "R is the lingua franca of statistical research. Work in all other languages should be discouraged." (Jan de Leeuw , 2003)
- "This is R. There is no if. Only how." (Simon 'Yoda' Blomberg, 2005)
- "Overall, SAS is about 11 years behind R and S-Plus in statistical capabilities (last year it was about 10 years behind) in my estimation." (Frank Harrell, 2003)
- "I quit using SAS in 1991 because my productivity jumped at least 20% within one month of using S-Plus." (Frank Harrell, 2003)

Taken from the R.-fortunes (selections from the R.-help list serve)

More fortunes

"You must realize that R is written by experts in statistics and statistical computing who, despite popular opinion, do not believe that everything in SAS and SPSS is worth copying. Some things done in such packages, which trace their roots back to the days of punched cards and magnetic tape when fitting a single linear model may take several days because your first 5 attempts failed due to syntax errors in the JCL or the SAS code, still reflect the approach of "give me every possible statistic that could be calculated from this model, whether or not it makes sense". The approach taken in R is different. The underlying assumption is that the useR is thinking about the analysis while doing it. " (Douglas Bates, 2007)

R is open source, how can you trust it?

- Q: "When you use it [R], since it is written by so many authors, how do you know that the results are trustable?"
- A: "The R engine [...] is pretty well uniformly excellent code but you have to take my word for that. Actually, you don't. The whole engine is open source so, if you wish, you can check every line of it. If people were out to push dodgy software, this is not the way they'd go about it." (Bill Venables, 2004
- "It's interesting that SAS Institute feels that non-peer-reviewed software with hidden implementations of analytic methods that cannot be reproduced by others should be trusted when building aircraft engines." – Frank Harrell (in response to the statement of the SAS director of technology product marketing: "We have customers who build engines for aircraft. I am happy they are not using freeware when I get on a jet.") R-help (January 2009)

What is R?: Technically

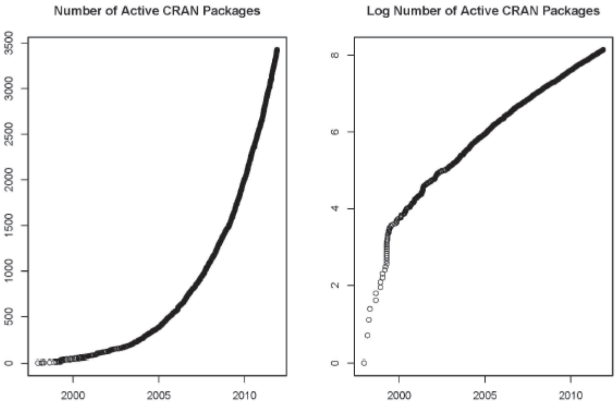
- R is an open source implementation of S (S-Plus is a commercial implementation)
- R is available under GNU Copy-left
- The current version of R is 3.02
- R is a group project run by a core group of developers (with new releases $\approx$ semiannually)
- R 3.1.0 is to be released sometime in 2014

(Adapted from Robert Gentleman)

R: A brief history

- 1991-93: Ross Dhaka and Robert Gentleman begin work on R project at U. Auckland
- 1995: R available by ftp under the GPL
- 96-97: mailing list and R core group are formed
- 2000: John Chambers, designer of S joins the Rcore (wins a prize for best software from ACM for S)
- 2001-2011: Core team continues to improve base package with a new release every 6 months.
- Many others contribute "packages" to supplement the functionality for particular problems
    - 2003-04-01: 250 packages
    - 2004-10-01: 500 packages
    - 2007-04-12: 1,000 packages
    - 2009-10-04: 2,000 packages
    - 2011-05-12 3,000 packages
    - 2012-08-23 4,000 packages
    - 2013-11-08 5,000 packages

Has R grown too much? Exponential growth rate continues



See also `http://r4stats.com/articles/popularity/`

Misconception: R is hard to use

1. R doesn't have a GUI (Graphical User Interface)
   - Partly true, many use syntax
   - Partly not true, GUIs exist (e.g., R Commander, R-Studio)
   - Quasi GUIs for Mac and PCs make syntax writing easier
2. R syntax is hard to use
   - Not really, unless you think an iPhone is hard to use
   - Easier to give instructions of 1-4 lines of syntax rather than pictures of what menu to pull down.
   - Keep a copy of your syntax, modify it for the next analysis.
3. R is not user friendly: A personological description of R
   - R is introverted: it will tell you what you want to know if you ask, but not if you don't ask.
   - R is conscientious: it wants commands to be correct.
   - R is not agreeable: its error messages are at best cryptic.
   - R is stable: it does not break down under stress.
   - R is open: new ideas about statistics are easily developed.

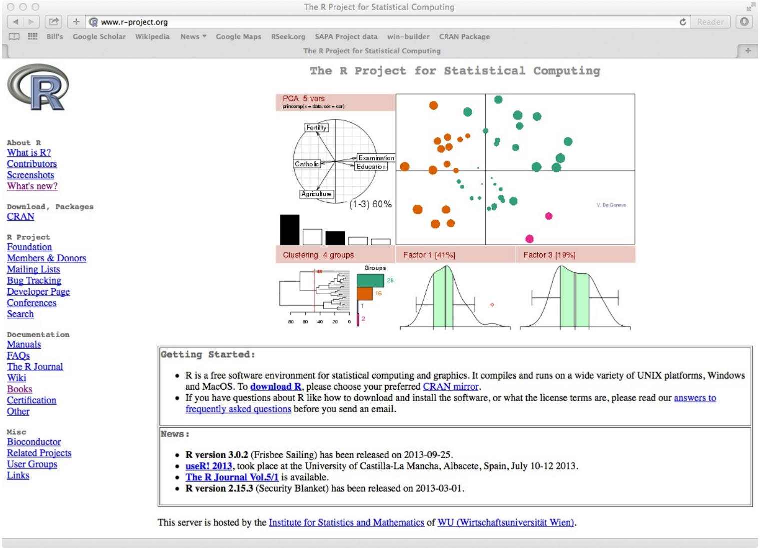
Misconceptions: R is hard to learn

1. With a brief web based tutorial `http://personality-project.org/r`, 2nd and 3rd year undergraduates in psychological methods and personality research courses are using R for descriptive and inferential statistics and producing publication quality graphics.
2. More and more psychology departments are using it for graduate and undergraduate instruction.
3. R is easy to learn, hard to master
   - R-help newsgroup is very supportive
   - Multiple web based and pdf tutorials see (e.g., `http://www.r-project.org/`)
   - Short courses using R for many applications
4. Books and websites for SPSS and SAS users trying to learn R (e.g.,`http://oit.utk.edu/scc/RforSAS&SPSSusers.pdf` by Bob Muenchen).

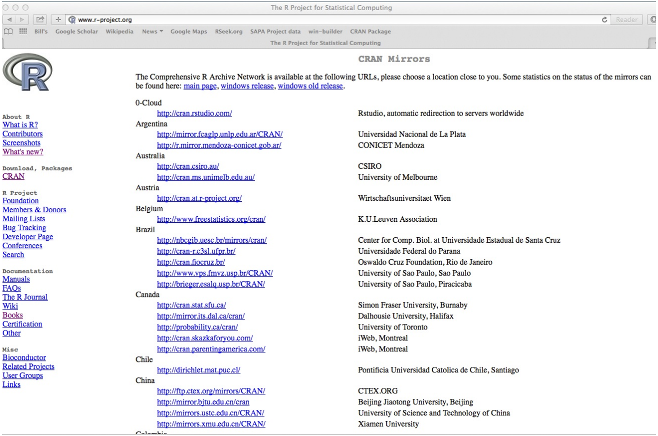
Ok, how do I get it: Getting started with R

1. Download from R Cran (`http://cran.r-project.org/`)
    - Choose appropriate operating system and download compiled R
2. Install R (current version is 3.02) with 3.1.0 coming this spring
3. Start R
4. Add useful packages (just need to do this once)
    - install.packages("ctv") #this downloads the task view package
    - library(ctv) #this activates the ctv package
    - install.views("Psychometrics") #among others
    - Take a 5 minute break
5. Activate the package(s) you want to use today (e.g., *psych*)
    - library(psych) #necessary for most of today's examples
    - library(sem) #will be used for a few examples
6. Use R
7. (See detailed tutorial at `https://personality-project.org/r/r.guide.html#gettingstarted`

Go to the R.project.org



Choose a mirror site near you


The R Project for Statistical Computing
www.r-project.org
Bill's Google Scholar Wikipedia News Google Maps RSeek.org SAPA Project data win-builder CRAN Package
The R Project for Statistical Computing

About R
What is R?
Contributors
Screenshots
What's new?

Download, Packages
CRAN

R Project
Foundation
Members & Donors
Mailing Lists
Bug Tracking
Developer Page
Conferences
Search

Documentation
Manuals
FAQs
The R Journal
Wiki
Books
Certification
Other

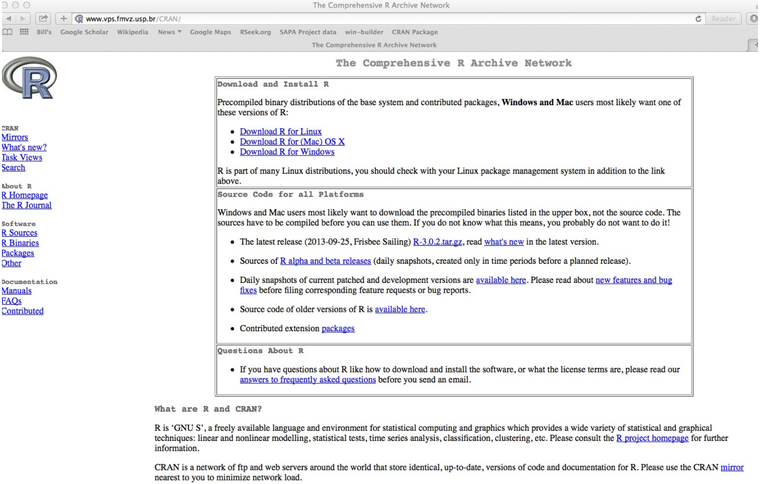
Misc
Bioconductor
Related Projects
User Groups
Links

CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: main page, windows release, windows old release.

0-Cloud	
http://cran.rstudio.com/	Rstudio, automatic redirection to servers worldwide
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
http://r.mirror.mendoza-conicet.gob.ar/	CONICET Mendoza
Australia	
http://cran.csiro.au/	CSIRO
http://cran.ms.unimelb.edu.au/	University of Melbourne
Austria	
http://cran.at.r-project.org/	Wirtschaftsuniversitaet Wien
Belgium	
http://www.freestatistics.org/cran/	K.U.Leuven Association
Brazil	
http://nbcgib.uesc.br/mirrors/cran/	Center for Comp. Biol. at Universidade Estadual de Santa Cruz
http://cran-r.c3sl.ufpr.br/	Universidade Federal do Parana
http://cran.fiocruz.br/	Oswaldo Cruz Foundation, Rio de Janeiro
http://www.vps.fmvz.usp.br/CRAN/	University of Sao Paulo, Sao Paulo
http://brieger.esalq.usp.br/CRAN/	University of Sao Paulo, Piracicaba
Canada	
http://cran.stat.sfu.ca/	Simon Fraser University, Burnaby
http://mirror.its.dal.ca/cran/	Dalhousie University, Halifax
http://probability.ca/cran/	University of Toronto
http://cran.skazkaforyou.com/	iWeb, Montreal
http://cran.parentingamerica.com/	iWeb, Montreal
Chile	
http://dirichlet.mat.puc.cl/	Pontificia Universidad Catolica de Chile, Santiago
China	
http://ftp.ctex.org/mirrors/CRAN/	CTEX.ORG
http://mirror.bjtu.edu.cn/cran	Beijing Jiaotong University, Beijing
http://mirrors.ustc.edu.cn/CRAN/	University of Science and Technology of China
http://mirrors.xmu.edu.cn/CRAN/	Xiamen University

Go to the Comprehensive R Archive Network (CRAN)



The Comprehensive R Archive Network

www.vps.fmvz.usp.br/CRAN/

Bill's Google Scholar Wikipedia News Google Maps RSeek.org SAPA Project data win-builder CRAN Package

The Comprehensive R Archive Network

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2013-09-25, Frisbee Sailing) R-3.0.2.tar.gz, read what's new in the latest version.
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.
- Source code of older versions of R is available here.
- Contributed extension packages

Questions About R

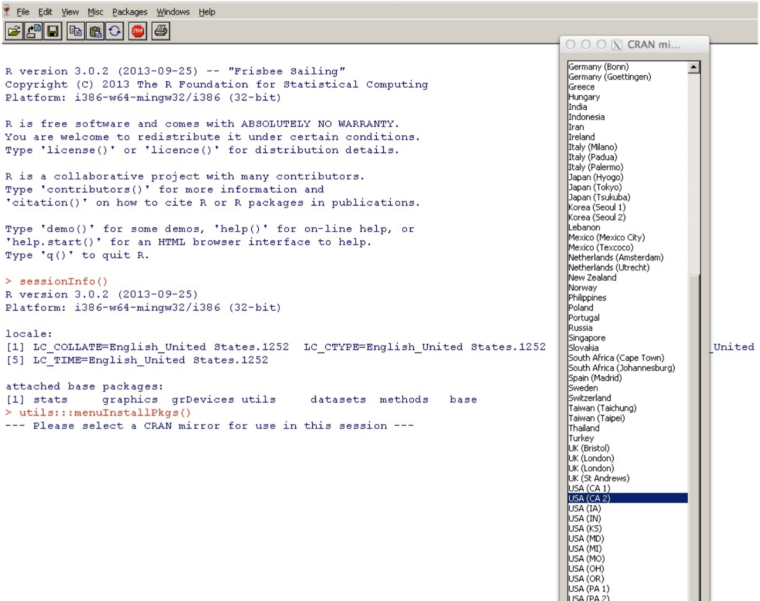
- If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

What are R and CRAN?

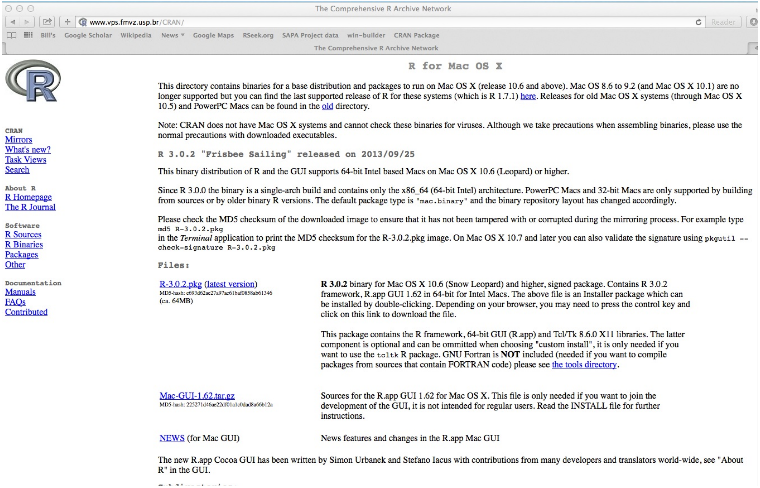
R is 'GNU S', a freely available language and environment for statistical computing and graphics which provides a wide variety of statistical and graphical techniques: linear and nonlinear modelling, statistical tests, time series analysis, classification, clustering, etc. Please consult the R project homepage for further information.

CRAN is a network of ftp and web servers around the world that store identical, up-to-date, versions of code and documentation for R. Please use the CRAN mirror nearest to you to minimize network load.

Download and install the appropriate version – PC



Download and install the appropriate version – Mac


The Comprehensive R Archive Network
www.vps.fmvz.usp.br/CRAN/
Bill's Google Scholar Wikipedia News Google Maps RSeek.org SAPA Project data win-builder CRAN Package
The Comprehensive R Archive Network

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

R for Mac OS X

This directory contains binaries for a base distribution and packages to run on Mac OS X (release 10.6 and above). Mac OS 8.6 to 9.2 (and Mac OS X 10.1) are no longer supported but you can find the last supported release of R for these systems (which is R 1.7.1) here. Releases for old Mac OS X systems (through Mac OS X 10.5) and PowerPC Macs can be found in the old directory.

Note: CRAN does not have Mac OS X systems and cannot check these binaries for viruses. Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 3.0.2 "Frisbee Sailing" released on 2013/09/25

This binary distribution of R and the GUI supports 64-bit Intel based Macs on Mac OS X 10.6 (Leopard) or higher.

Since R 3.0.0 the binary is a single-arch build and contains only the x86_64 (64-bit Intel) architecture. PowerPC Macs and 32-bit Macs are only supported by building from sources or by older binary R versions. The default package type is "mac.binary" and the binary repository layout has changed accordingly.

Please check the MD5 checksum of the downloaded image to ensure that it has not been tampered with or corrupted during the mirroring process. For example type
md5 R-3.0.2.pkg
in the *Terminal* application to print the MD5 checksum for the R-3.0.2.pkg image. On Mac OS X 10.7 and later you can also validate the signature using pkgutil --check-signature R-3.0.2.pkg

Files:

R-3.0.2.pkg (latest version)
MD5-hash: e693d62ac27a97ac61baf0858ab61346
(ca. 64MB)

R 3.0.2 binary for Mac OS X 10.6 (Snow Leopard) and higher, signed package. Contains R 3.0.2 framework, R.app GUI 1.62 in 64-bit for Intel Macs. The above file is an Installer package which can be installed by double-clicking. Depending on your browser, you may need to press the control key and click on this link to download the file.

This package contains the R framework, 64-bit GUI (R.app) and Tcl/Tk 8.6.0 X11 libraries. The latter component is optional and can be ommitted when choosing "custom install", it is only needed if you want to use the tcltk R package. GNU Fortran is **NOT** included (needed if you want to compile packages from sources that contain FORTRAN code) please see the tools directory.

Mac-GUI-1.62.tar.gz
MD5-hash: 225271d46ae22df01a1c0dad8a66b12a

Sources for the R.app GUI 1.62 for Mac OS X. This file is only needed if you want to join the development of the GUI, it is not intended for regular users. Read the INSTALL file for further instructions.

NEWS (for Mac GUI)

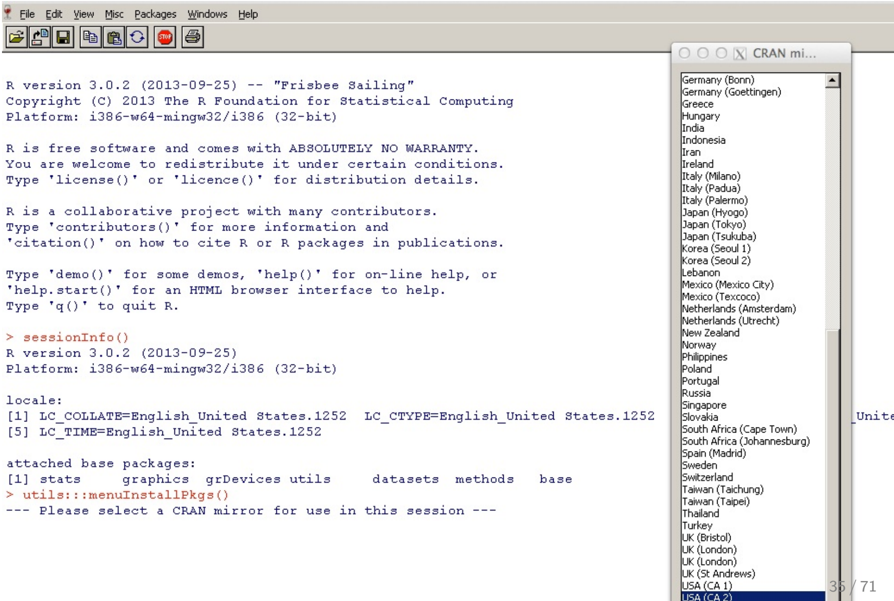
News features and changes in the R.app Mac GUI

The new R.app Cocoa GUI has been written by Simon Urbanek and Stefano Iacus with contributions from many developers and translators world-wide, see "About R" in the GUI.

Starting R on a PC

```
R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Starting R on a PC



Start up R and get ready to play (Mac version)

```
R Under development (unstable) (2014-01-26 r64896) -- "Unsuffered Consequences"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin10.8.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.62 (6623) x86_64-apple-darwin10.8.0]

[Workspace restored from /Users/revelle/.RData]
[History restored from /Users/revelle/.Rapp.history]
```

Annotated installation guide: don't type the >

```
> install.packages("ctv")

> library(ctv)

> install.views("Psychometrics")

#or just install a few packages
> install.packages("psych")

> install.packages("GPArotation")
> install.packages("MASS")
> install.packages("mvtnorm")
```

- Install the task view installer package. You might have to choose a "mirror" site.
- Make it active
- Install all the packages in the "Psychometrics" task view. This will take a few minutes.
- Or, just install one package (e.g., psych)
- as well as a few suggested packages that add functionality for factor rotation, multivariate normal distributions, etc.
-

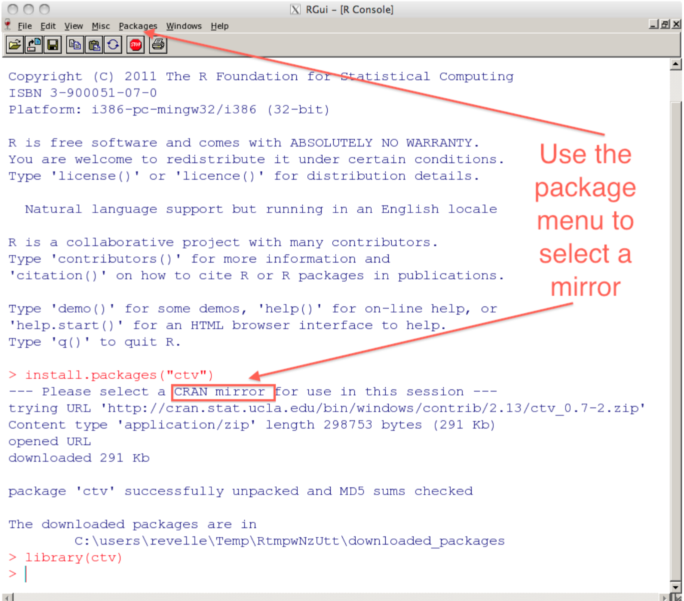
Installing just the psych package

RGui - [R Console]

File Edit View Misc Packages Windows Help

```
R version 2.13.0 (2011-04-13)
Copyright (C) 2011 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: i386-pc-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> install.packages("psych")
--- Please select a CRAN mirror for use in this session ---
trying URL 'http://cran.stat.ucla.edu/bin/windows/contrib/2.13/psych_1.0-97.zip'
Content type 'application/zip' length 1952216 bytes (1.9 Mb)
opened URL
downloaded 1.9 Mb
```

Or, install and use ctv package to load a task view on a PC



Check the version number for R (should be $\geq$ 3.02 and for psych ($\geq$1.4.2)

```
> library(psych)
> sessionInfo()

R Under development (unstable) (2014-01-26 r64896)
Platform: x86_64-apple-darwin10.8.0 (64-bit)

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] psych_1.4.2
```

R is extensible: The use of "packages"

- More than 5000 packages are available for R (and growing daily)
- Can search all packages that do a particular operation by using the sos package
  - `install.packages("sos")` #if you haven't already
  - library(sos) # make it active once you have it
    - `findFn`("X") #will search a web data base for all packages/functions that have "X"
    - `findFn`("principal components analysis ") #will return 1918 matches and reports the top 400 and download 364 links to 129 packages
    - `findFn`("Item Response Theory") # will return 310 matches with 260 links in 47 packages
    - `findFn`("INDSCAL ") # will return 7 matches.
- install.packages("X") will install a particular package (add it to your R library – you need to do this just once)
- library(X) #will make the package X available to use if it has been installed (and thus in your library)

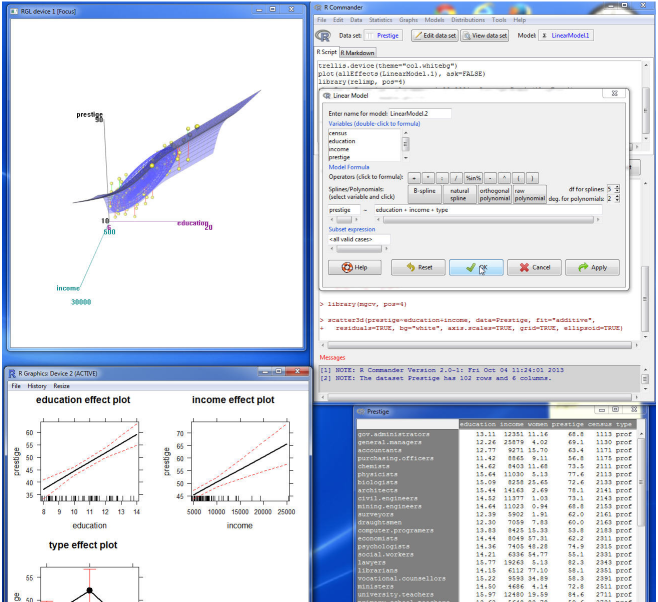
A small subset of very useful packages

- General use
  - core R
  - MASS
  - lattice
  - lme4 (core)
  - psych
  - Zelig
- Special use
  - ltm
  - sem
  - lavaan
  - OpenMx
  - GPArotation
  - mvtnorm
  - $>$ 5000 known
  - $+$?
- General applications
  - most descriptive and inferential stats
  - Modern Applied Statistics with S
  - Lattice or Trellis graphics
  - Linear mixed-effects models
  - Personality and psychometrics
  - General purpose toolkit
- More specialized packages
  - Latent Trait Model (IRT)
  - SEM and CFA (multiple groups)
  - SEM and CFA (multiple groups)
  - SEM and CFA (multiple groups +)
  - Jennrich + Browne rotations
  - Multivariate distributions
  - Thousands of more packages on CRAN
  - Code on webpages/journal articles

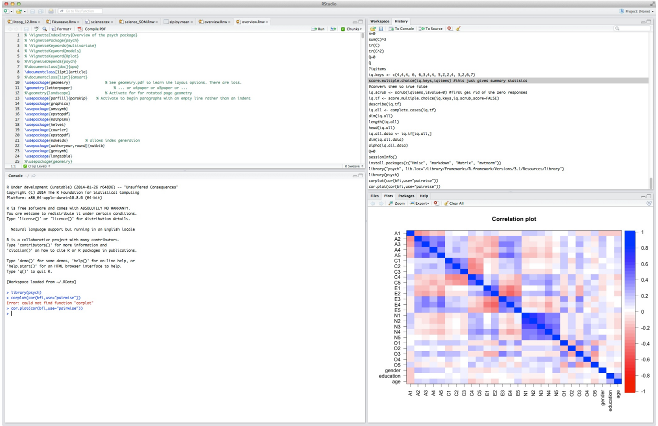
Implementations

1. Base R in the Unix/Linux/Mac X11 framework
   -
2. Base R on the Mac/PC
   - Mac has prompts at bottom of window
3. Graphical User Interfaces
   - R Commander
   - R studio as a convenient shell

R Commander (by John Fox) has a basic GUI



RStudio (particularly nice for PCs)



Using R

1. Install the relevant packages (just once!)
    - Either one at a time, or by using a "task view"
2. Make the packages you want to use "active" by library(package name) e.g., library(psych)
    - For each session
    - Can be automatized
3. Use the functions in a package
    - To see all functions in a package go to the index of the package or use the objects function: e.g., objects(package:psych)
    - Apply a function to data
    - All functions require an object to act upon. Most require this in parentheses. All functions return an object. This may be saved for later.
        - function(object) #apply the function to the object, show result
        - sqrt(2)
        - result <- function(object) #apply the function to object, save result
        - answer <- alpha(ability) #lots and lots of output is saved

Basic R commands – remember don't enter the >

R is just a fancy calculator. Add, subtract, sum, products, group

```
> 2 + 2
[1] 4
> 3^4
[1] 81
> sum(1:10)
[1] 55
> prod(c(1, 2, 3, 5, 7))
[1] 210
```

It is also a statistics table (the normal distribution, the t distribution, F, χ^2, ...)

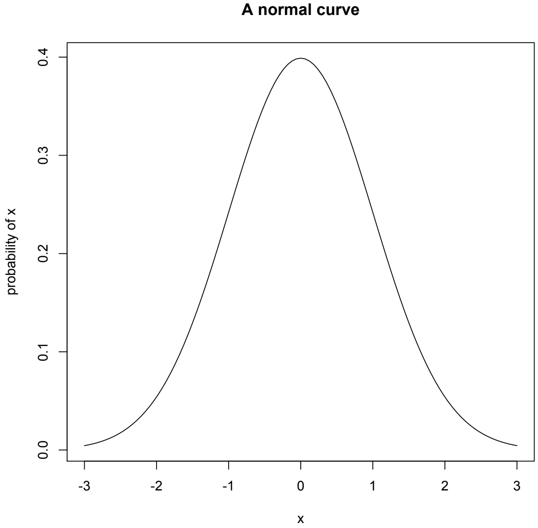
```
> pnorm(q = 1)  #probability of normal value  > 1
[1] 0.8413447
> pt(q = 2, df = 20) #probability of t  > 2 with 20 df
[1] 0.9703672
```

R is a set of distributions. Don't buy a stats book with tables!

Table : To obtain the density, prefix with *d*, probability with *p*, quantiles with *q* and to generate random values with *r*. (e.g., the normal distribution may be chosen by using dnorm, pnorm, qnorm, or rnorm.)

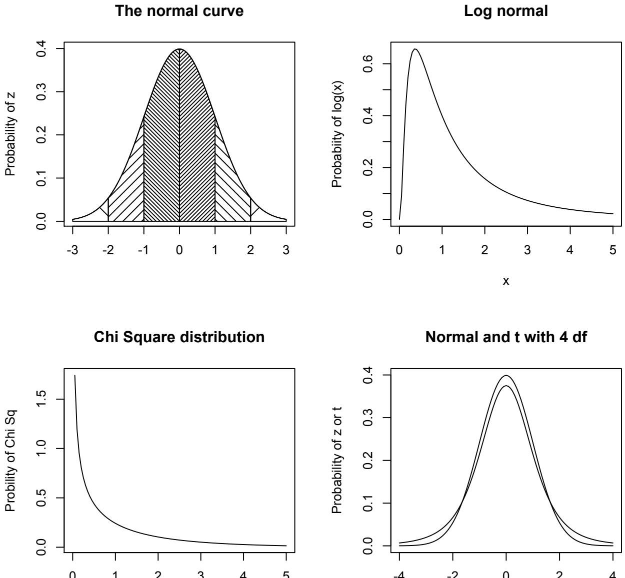
Distribution	base name	P 1	P 2	P 3	example application
Normal	norm	mean	sigma		Most data
Multivariate normal	mvnorm	mean	r	sigma	Most data
Log Normal	lnorm	log mean	log sigma		income or reaction time
Uniform	unif	min	max		rectangular distributions
Binomial	binom	size	prob		Bernuilli trials (e.g. coin flips)
Student's t	t	df		nc	Finding significance of a t-test
Multivariate t	mvt	df	corr	nc	Multivariate applications
Fisher's F	f	df1	df2	nc	Testing for significance of F test
χ^2	chisq	df		nc	Testing for significance of χ^2
Exponential	exp	rate			Exponential decay
Gamma	gamma	shape	rate	scale	distribution theoryh
Hypergeometric	hyper	m	n	k	
Logistic	logis	location	scale		Item Response Theory
Poisson	pois	lambda			Count data
Weibull	weibull	shape	scale		Reaction time distributions
Cauchy	cauchy	location	scale	log	infinite variance!

R can draw distributions



curve(dnorm(x),-3,3,ylab="probability of x",main="A normal curve")

R can draw more interesting distributions



R is also a graphics calculator

The first line draws the normal curve, the second prints the title, the next lines draw the cross hatching.

```
op <- par(mfrow=c(2,2))       #set up a 2 x 2 graph
curve(dnorm(x),-3,3,xlab="",ylab="Probability of z")
title(main="The normal curve",outer=FALSE)
xvals <-  seq(-3,-2,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=2,angle=-45)
xvals <-  seq(-2,-1,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=14,angle=45)
xvals <-  seq(-1,-0,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=34,angle=-45)
xvals <-  seq(2,3,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=2,angle=45)
xvals <-  seq(1,2,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=14,angle=-45)
xvals <-  seq(0,1,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=34,angle=45)
curve(dlnorm(x),0,5,ylab='Probabiity of log(x)',main='Log normal')
curve(dchisq(x,1),0,5,ylab='Probility of Chi Sq',xlab='Chi Sq',main='Chi Square distribution')
curve(dnorm(x),-4,4,ylab='Probability of z or t',xlab='z or t',main='Normal and t with 4 df')
curve(dt(x,4),add=TRUE)
op <- par(mfrow=c(1,1))
```

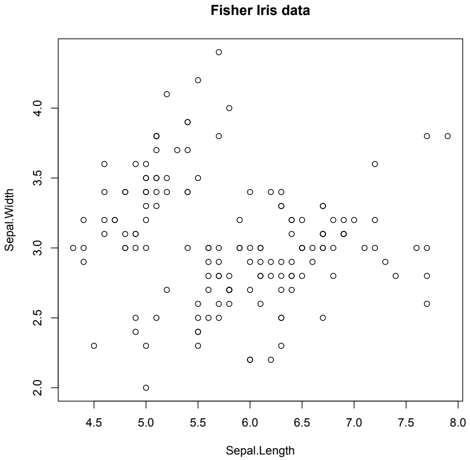
Example data sets built into many packages

Table : Some of the 48 data sets in the psych package

Name	Content description
veg	Thurstone's Vegetables
cities	Airplane distances for 11 US cities
galton	Francis Galton's original data set of heights
cushny	The original t-test data from "student" (Gossett)
ability	16 ability items from SAPA
bfi	25 Big Five items + gender, age, education from SAPA
sat.act	Test scores, gender, age and education
Thurstone	9 ability variables from Thurstone
msq	75 mood items from the PMC lab
neo	Correlation matrix of the 30 NEO-PI-R facets

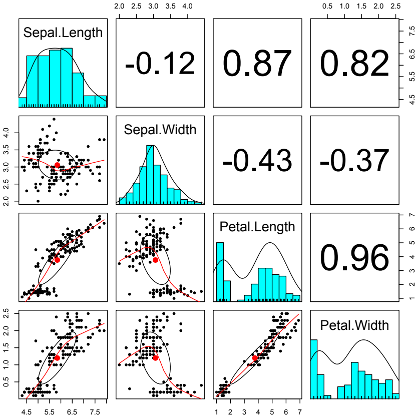
```
data() #to see all available
data(package="psych")  # to see  all psych data sets
```

A simple scatter plot using `plot` shows Fisher's Iris data set




```
plot(iris[1:2],xlab="Sepal.Length",ylab="Sepal.Width"
,main="Fisher Iris data")
```

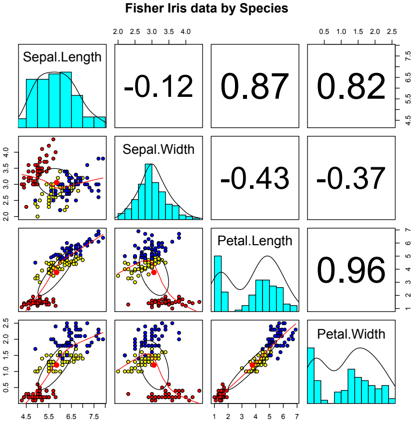
A scatter plot matrix with loess regression using `pairs.panels` shows more information than a simple scatter plot




```
pairs.panels(iris[1:4])
```

1. Correlations above the diagonal
2. Diagonal shows histograms and densities
3. scatter plots below the diagonal with correlation ellipse
4. locally smoothed (loess) regressions for each pair

A better SPLOM with colors for groups using pairs.panels



1. Correlations above the diagonal
2. Diagonal shows histograms and densities
3. scatter plots below the diagonal with correlation ellipse
4. locally smoothed (loess) regressions for each pair
5. optional color coding of grouping variables.

```
pairs.panels(iris[1:4],bg=c("red","yellow","blue")
[iris$Species],pch=21,main="Fisher Iris data by
Species")
```

Using R for psychological statistics: Basic statistics

1. Writing syntax
   - For a single line, just type it
   - Mistakes can be redone by using the up arrow key
   - For longer code, use a text editor (built into some GUIs)
2. Data entry
   - Using built in data sets for examples
   - Copying from another program (using the "clipboard")
   - Reading a text or csv file
   - Importing from SPSS or SAS
   - Simulate it (using various simulation routines)
3. Descriptives
   - Graphical displays
   - Descriptive statistics
   - Correlation
4. Inferential
   - the t test
   - the F test
   - the linear model

Data entry overview

1. Using built in data sets for examples
   - `data()` will list $>$ 100 data sets in the `datasets` package as well as all sets in loaded packages.
   - Most packages have associated data sets used as examples
   - *psych* has $>$ 40 example data sets
2. Copying from another program
   - use copy and paste into R using `read.clipboard` and its variations
3. Reading a text or csv file
   - read a local or remote file
4. Importing from SPSS or SAS
5. Simulate it (using various simulation routines)

Examples of built in data sets from the psych package

```
> data(package="psych")
Bechtoldt                Seven data sets showing a bifactor solution.
Dwyer                    8 cognitive variables used by Dwyer for an examp
Reise                    Seven data sets showing a bifactor solution.
all.income (income)      US family income from US census 2008
bfi                      25 Personality items representing 5 factors
blot                     Bond's Logical Operations Test - BLOT
burt                     11 emotional variables from Burt (1915)
cities                   Distances between 11 US cities
epi.bfi                  13 personality scales from the Eysenck Personali
                         and Big 5 inventory
affect                   Two data sets of affect and arousal scores as a
                         personality and movie conditions.
galton                   Galton's Mid parent child height data
income                   US family income from US census 2008
iqitems                  16 multiple choice IQ items
msq                      75 mood items from the Motivational State Questi
                         for N = 3896
neo                      NEO correlation matrix from the NEO_PI_R manual
sat.act                  3 Measures of ability: SATV, SATQ, ACT
Thurstone                The classic Thurstone 9 variable problem
veg (vegetables)         Paired comparison of preferences for 9 vegetable
```

Reading data from another program –using the clipboard

1. Read the data in your favorite spreadsheet or text editor
2. Copy to the clipboard
3. Execute the appropriate `read.clipboard` function with or without various options specified

```
my.data <- read.clipboard()   #assumes headers and tab or space delimited
my.data <- read.clipboard.csv()   #assumes headers and comma delimited
my.data <- read.clipboard.tab()   #assumes headers and tab delimited
                                          (e.g., from Excel)
my.data <- read.clipboard.lower()  #read in a matrix given the lower
my.data <- read.clipboard.upper() #  or upper  off diagonal
my.data <- read.clipboard.fwf()  #read in data using a fixed format width
                                            (see read.fwf for instructions)
```

4. `read.clipboard()` has default values for the most common cases and these do not need to be specified. Consult ?read.clipboard for details.

Reading from a local or remote file

1. Perhaps the standard way of reading in data is using the `read` command.
   - First must specify the location of the file
   - Can either type this in directly or use the `file.choose` function
   - The file name/location can be a remote URL
2. Two examples of reading data

```
file.name <- file.choose() #this opens a window to allow you find the file
my.data <- read.table(file.name)
datafilename="http://personality-project.org/r/datasets/R.appendix1.data"
data.ex1=read.table(datafilename,header=TRUE)   #read the data into a table
```

```
> dim(data.ex1)  #what are the dimensions of what we read?
[1] 18  2
> describe(data.ex1) #do the data look right?
          var  n  mean   sd median trimmed  mad min max range skew kurtosi
Dosage*     1 18  1.89 0.76      2    1.88 1.48   1   3     2 0.16    -1.1
Alertness   2 18 27.67 6.82     27   27.50 8.15  17  41    24 0.25    -0.6
```

read a “foreign” file e.g., an SPSS sav file

`read.spss` reads a file stored by the SPSS save or export commands.

```
read.spss(file, use.value.labels = TRUE, to.data.frame = FALSE,
          max.value.labels = Inf, trim.factor.names = FALSE,
          trim_values = TRUE, reencode = NA, use.missings = to.data.frame)
```

file	Character string: the name of the file or URL to read.
use.value.labels	Convert variables with value labels into R factors with those levels?
to.data.frame	return a data frame? Defaults to FALSE, probably should be TRUE in most cases.
max.value.labels	Only variables with value labels and at most this many unique values will be converted to factors if use.value.labels = *TRUE*.
trim.factor.names	Logical: trim trailing spaces from factor levels?
trim_values	logical: should values and value labels have trailing spaces ignored when matching for use.value.labels = *TRUE*?
use.missings	logical: should information on user-defined missing values be used to set the corresponding values to NA?

Simulate data

For many demonstration purposes, it is convenient to generate simulated data with a certain defined structure. The *psych* package has a number of built in simulation functions. Here are a few of them.

1. Simulate various item structures
   - sim.congeneric A one factor congeneric measure model
   - sim.items A two factor structure with either simple structure or a circumplex structure.
   - sim.rasch Generate items for a one parameter IRT model.
   - sim.irt Generate items for a one-four parameter IRT Model
2. Simulate various factor structures
   - sim.simplex Default is a four factor structure with a three time point simplex structure.
   - sim.hierarchical Default is 9 variables with three correlated factors.

Get the data and look at it

Read in some data, look at the first and last few cases, and then get basic descriptive statistics. For this example, we will use a built in data set.

```
> my.data <- epi.bfi
> headtail(my.data)
```

	epiE	epiS	epiImp	epilie	epiNeur	bfagree	bfcon	bfext	bfneur	bfopen	bdi	traitanx	stateanx
1	18	10	7	3	9	138	96	141	51	138	1	24	22
2	16	8	5	1	12	101	99	107	116	132	7	41	40
3	6	1	3	2	5	143	118	38	68	90	4	37	44
4	12	6	4	3	15	104	106	64	114	101	8	54	40
...	...	...	...	...	...	...	...	...	...	...	...	...	...
228	12	7	4	3	15	155	129	127	88	110	9	35	34
229	19	10	7	2	11	162	152	163	104	164	1	29	47
230	4	1	1	2	10	95	111	75	123	138	5	39	58
231	8	6	3	2	15	85	62	90	131	96	24	58	58

epi.bfi has 231 cases from two personality measures

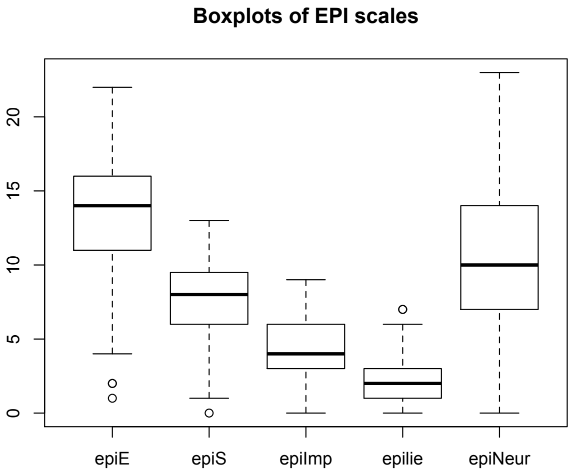
Now find the descriptive statistics for this data set

```
> describe(my.data)

         var   n   mean    sd median trimmed   mad min max range  skew kurtosis   se
epiE       1 231  13.33  4.14     14   13.49  4.45   1  22    21 -0.33    -0.01 0.27
epiS       2 231   7.58  2.69      8    7.77  2.97   0  13    13 -0.57     0.04 0.18
epiImp     3 231   4.37  1.88      4    4.36  1.48   0   9     9  0.06    -0.59 0.12
epilie     4 231   2.38  1.50      2    2.27  1.48   0   7     7  0.66     0.30 0.10
epiNeur    5 231  10.41  4.90     10   10.39  4.45   0  23    23  0.06    -0.46 0.32
bfagree    6 231 125.00 18.14    126  125.26 17.79  74 167    93 -0.21    -0.22 1.19
bfcon      7 231 113.25 21.88    114  113.42 22.24  53 178   125 -0.02     0.29 1.44
bfext      8 231 102.18 26.45    104  102.99 22.24   8 168   160 -0.41     0.58 1.74
bfneur     9 231  87.97 23.34     90   87.70 23.72  34 152   118  0.07    -0.51 1.54
bfopen    10 231 123.43 20.51    125  123.78 20.76  73 173   100 -0.16    -0.11 1.35
bdi       11 231   6.78  5.78      6    5.97  4.45   0  27    27  1.29     1.60 0.38
traitanx  12 231  39.01  9.52     38   38.36  8.90  22  71    49  0.67     0.54 0.63
stateanx  13 231  39.85 11.48     38   38.92 10.38  21  79    58  0.72     0.04 0.76
```

Boxplots are a convenient descriptive device

Show the Tukey "boxplot" for the Eysenck Personality Inventory

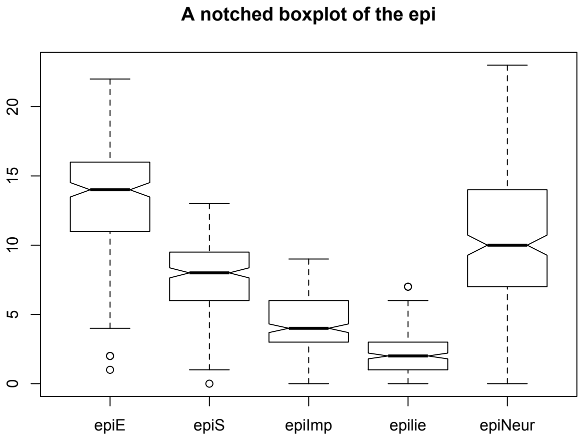
```
boxplot(my.data[1:5]) #just the first 5 variables
```



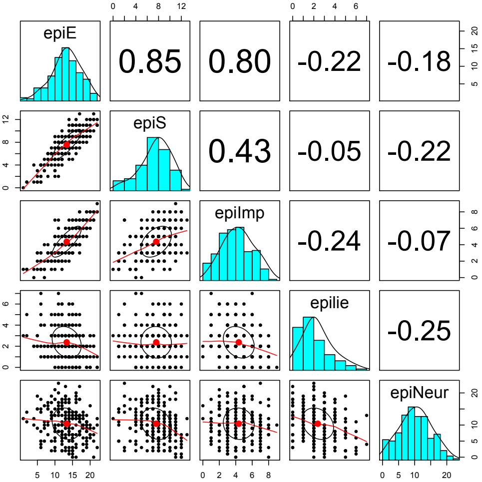
Enhanced box plots are even more convenient descriptive devices

Show the Tukey "boxplot" for the Eysenck Personality Inventory

```
boxplot(my.data[1:5]) #just the first 5 variables
```



Plot the scatter plot matrix (SPLOM) of the first 5 variables using the `pairs.panels`function



Use the `pairs.panels` function from *psych*

```
pairs.panels(my.data[1:5])
```

Find the correlations for this data set, round off to 2 decimal places

```
> round(cor(my.data, use = "pairwise"), 2)

          epiE  epiS epiImp epilie epiNeur bfagree bfcon bfext bfneur bfopen
epiE      1.00  0.85   0.80  -0.22   -0.18    0.18 -0.11  0.54  -0.09   0.14 -0
epiS      0.85  1.00   0.43  -0.05   -0.22    0.20  0.05  0.58  -0.07   0.15 -0
epiImp    0.80  0.43   1.00  -0.24   -0.07    0.08 -0.24  0.35  -0.09   0.07 -0
epilie   -0.22 -0.05  -0.24   1.00   -0.25    0.17  0.23 -0.04  -0.22  -0.03 -0
epiNeur  -0.18 -0.22  -0.07  -0.25    1.00   -0.08 -0.13 -0.17   0.63   0.09  0
bfagree   0.18  0.20   0.08   0.17   -0.08    1.00  0.45  0.48  -0.04   0.39 -0
bfcon    -0.11  0.05  -0.24   0.23   -0.13    0.45  1.00  0.27   0.04   0.31 -0
bfext     0.54  0.58   0.35  -0.04   -0.17    0.48  0.27  1.00   0.04   0.46 -0
bfneur   -0.09 -0.07  -0.09  -0.22    0.63   -0.04  0.04  0.04   1.00   0.29  0
bfopen    0.14  0.15   0.07  -0.03    0.09    0.39  0.31  0.46   0.29   1.00 -0
bdi      -0.16 -0.13  -0.11  -0.20    0.58   -0.14 -0.18 -0.14   0.47  -0.08  1
traitanx -0.23 -0.26  -0.12  -0.23    0.73   -0.31 -0.29 -0.39   0.59  -0.11  0
stateanx -0.13 -0.12  -0.09  -0.15    0.49   -0.19 -0.14 -0.15   0.49  -0.04  0
```

Find the correlations using `lowerCor`

```
> lowerCor(my.data)

         epiE  epiS  epImp epili epiNr bfagr bfcon bfext bfner bfopn bdi   trtn
epiE      1.00
epiS      0.85  1.00
epiImp    0.80  0.43  1.00
epilie   -0.22 -0.05 -0.24  1.00
epiNeur  -0.18 -0.22 -0.07 -0.25  1.00
bfagree   0.18  0.20  0.08  0.17 -0.08  1.00
bfcon    -0.11  0.05 -0.24  0.23 -0.13  0.45  1.00
bfext     0.54  0.58  0.35 -0.04 -0.17  0.48  0.27  1.00
bfneur   -0.09 -0.07 -0.09 -0.22  0.63 -0.04  0.04  0.04  1.00
bfopen    0.14  0.15  0.07 -0.03  0.09  0.39  0.31  0.46  0.29  1.00
bdi      -0.16 -0.13 -0.11 -0.20  0.58 -0.14 -0.18 -0.14  0.47 -0.08  1.00
traitanx -0.23 -0.26 -0.12 -0.23  0.73 -0.31 -0.29 -0.39  0.59 -0.11  0.65  1.0
stateanx -0.13 -0.12 -0.09 -0.15  0.49 -0.19 -0.14 -0.15  0.49 -0.04  0.61  0.5
```

A heat map of 25 BFI items using `cor.plot`